Found Comput Math (2015) 15:1213–1244
DOI 10.1007/s10208-014-9223-y

FOUNDATIONS OF COMPUTATIONAL MATHEMATICS
The Journal of the Society for the Foundations of Computational Mathematics

The Persistent Homology of a Self-Map

Herbert Edelsbrunner · Grzegorz Jabłoński · Marian Mrozek

Received: 20 May 2013 / Revised: 17 July 2014 / Accepted: 20 August 2014 /
Published online: 9 October 2014

Abstract Considering a continuous self-map and the induced endomorphism on homology, we study the eigenvalues and eigenspaces of the latter. Taking a filtration of representations, we define the persistence of the eigenspaces, effectively introducing a hierarchical organization of the map. The algorithm that computes this information for a finite sample is proved to be stable, and to give the correct answer for a sufficiently dense sample. Results computed with an implementation of the algorithm provide evidence of its practical utility.

Keywords Discrete dynamical systems · Computational topology · Persistent homology · Category theory · Algebraic algorithms · Convergence · Stability · Computational experiments

Mathematics Subject Classification 55N35 · 37B99 · 18A99 · 55U99

1 Introduction

In recent years, the theory of persistent homology [11,23] has become a useful tool in several areas, including shape analysis [12], scientific visualization [13], high-dimensional data analysis [3], but also in mathematics itself [21]. The specific aim of

Communicated by Konstantin Mischaikow.

H. Edelsbrunner
IST Austria (Institute of Science and Technology Austria), Klosterneuburg, Austria

G. Jabłoński (✉) · M. Mrozek
Division of Computational Mathematics, Faculty of Mathematics and Computer Science,
Jagiellonian University, Kraków, Poland
e-mail: grzegorz.jablonski@uj.edu.pl

this paper is an approach to the persistence of endomorphisms induced in homology by continuous self-maps. The long-term goal is to embed persistence in the computational analysis of dynamical systems, as pursued in [14] and the related literature.

In the case of finitely generated homology with field coefficients, the homomorphism induced by a continuous map between topological spaces is a linear map between finite-dimensional vector spaces. Such a map $\varphi : Y \to X$ is characterized up to conjugacy by its rank. This is in contrast to a linear self-map, $\phi : X \to X$, which in the case of an algebraically closed field[1] is characterized up to conjugacy by its Jordan form. A weaker piece of information is the eigenvectors, which in our setting captures the homology classes that are invariant under the self-map. Therefore, it is natural to study the persistence of eigenvalues and eigenspaces as a first step to the full understanding of the persistence of the map. We define it in terms of the persistence of vector spaces, a concept that has been around for some time. Specifically, it has been presented as the general idea of zigzag persistence [2], which is based on the theory of quivers [9]. Since we need an algorithm that provides not only the persistence intervals but also a special basis, we give an independent presentation of the concept. We believe this presentation is elementary and in the spirit of the theory of persistent homology. We also note that its generalization to zigzag persistence is straightforward. Beyond describing the algorithm for the persistence of eigenvalues and eigenspaces, we analyze its performance, proving that the persistence diagram it produces is stable under perturbations of the input, and the algorithm converges to the homology of the studied map, reaching the correct ranks for a sufficiently fine sample. In addition, we exhibit results obtained with a software implementation, which suggest that the persistent homology of eigenspaces picks up the important dynamics already from a relatively small sample.

We motivate the technical work in this paper with a small toy example, designed to highlight one of the main difficulties we encounter. Writing $\mathbb{S}^1$ for the circle in the complex plane, we consider the map defined by $f(z) := z^2$, which doubles the angle of the input point. Sampling $\mathbb{S}^1$ at eight equally spaced locations, $x_j := \cos \frac{j\pi}{4} + \mathbf{i} \sin \frac{j\pi}{4}$, we can check that $f(x_j) = x_{2j}$, where indices are taken modulo 8. Assuming the space and the map are unknown, other than at the sampled points and their images, we consider the filtration $K_1 \subseteq K_2 \subseteq K_3 \subseteq K_4 \subseteq K_5$ shown in the top row of Fig. 1. Each complex K_i consists of all simplices spanned by the eight points whose smallest enclosing circles have radii that do not exceed a given threshold, and this threshold increases from left to right. The most persistent homology classes in this filtration are the component that appears in K_1 and lasts to K_5 and the loop that appears in K_2 and lasts to K_4.

We would hope that f extends to simplicial maps on these complexes, but this is unfortunately not the case in general. For instance, f maps the endpoints of the edge x_1x_7 in K_3 to the points x_2 and x_6, but they are not endpoints of a common edge in K_3. The reason for this situation is the expanding character of f. To still make sense of the map, we construct the maximal partial simplicial maps, $\kappa_i : K_i \nrightarrow K_i$ consistent with f. Figure 1 shows the domains of these maps in the bottom row, and for $i = 3, 4$, we can see how κ_i wraps the convex octagon twice around the hole in K_i shown right

[1] A field is *algebraically closed* if every non-constant polynomial over the field has a root.

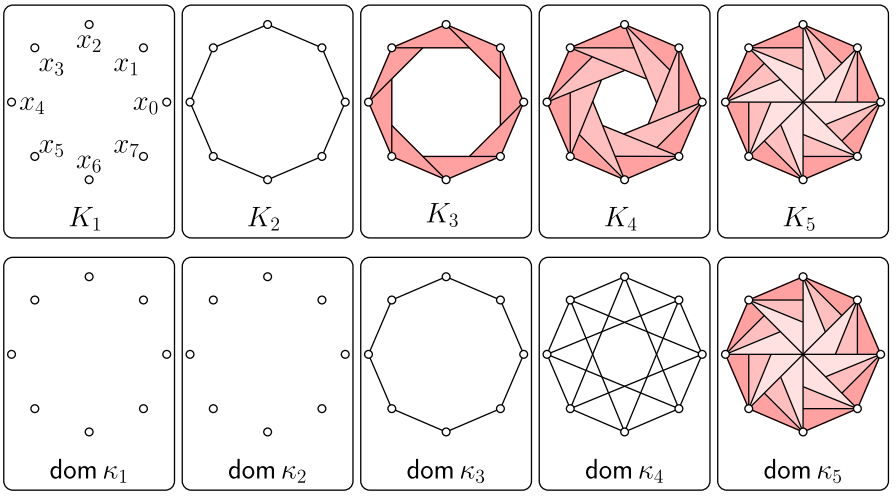

Fig. 1 Five simplicial complexes in the filtration of the eight data points at the *top*, and the domains of the induced partial simplicial maps at the *bottom*

above. This reflects the fundamental feature of f, namely that its image wraps around the circle twice. To see this more formally, we compare the homology classes in the domains with their images. For $i = 1, 3, 5$, the inclusion of the domain of κ_i in K_i induces an isomorphism in homology. The comparison therefore reduces to the study of eigenvectors of an endomorphism. The lack of isomorphism for $i = 2, 4$ may be overcome by the study of eigenvectors of pairs of linear maps. In this particular case, we are able to conclude that the eigenspace for eigenvalue $t = 2$ appears in K_3 and lasts to K_4, thus reconstructing the essential character of f from a very small sample.

To summarize, there are differences between the partial simplicial maps and the underlying continuous map; see in particular the reorganization that takes place at $i = 2$ and $i = 4$. The hope to recover the properties of the latter from the former is based on the ability of persistence to provide a measure that transcends fluctuations and identifies what stays the same when things change.

Outline Section 2 introduces the categories of partial functions, matchings, and linear maps. Section 3 discusses towers within these categories and introduces the concept of persistence. Section 4 describes the algorithm that computes the persistent homology of an endomorphism from a hierarchical representation of the underlying self-map. Section 5 proves that the algorithm converges and produces stable persistence diagrams. Section 6 presents results obtained with our implementation of the algorithm. Section 7 concludes the paper.

2 Categories

We find the language of category theory convenient to talk about persistent homology; see MacLane [16] for a standard reference. Most importantly, we introduce the

category of finite sets and matchings, which will lead to an elementary exposition of persistence.

2.1 Partial Functions

We recall that a *category* consists of *objects* and (directed) *arrows* between objects. Importantly, there is the *identity arrow* from every object to itself, and arrows compose associatively. An arrow, $\theta : K \to L$, is *invertible* if it has an inverse, $\theta^{-1} : L \to K$, such that $\theta^{-1}\theta$ and $\theta\theta^{-1}$ are the identity arrows for K and L. If there is an invertible arrow from K to L, then the two objects are *isomorphic*. Every category in this paper contains a *zero object*, which is characterized by having exactly one arrow to and one arrow from every other object. It is unique up to isomorphisms. Two arrows $\kappa : K \to K'$ and $\lambda : L \to L'$ are *conjugate* if there are invertible arrows $\theta : K \to L$ and $\theta' : K' \to L'$ that commute with κ and λ; that is, $\theta'\kappa = \lambda\theta$. A *functor* is an arrow between categories, assigning to each object and each arrow of the first category an object and an arrow of the second category in such a way that the identity arrows are mapped to identity arrows and the functor commutes with the composition of the arrows.

We use a category whose arrows generalize functions between sets as the basis of other categories. Specifically, a *partial function* is a relation $\xi \subseteq X \times Y$ such that every $x \in X$ is either not paired or paired with exactly one element in Y [15]. We denote it by $\xi : X \rightarrowtail Y$, observing that there is a largest subset $X' \subseteq X$ such that the restriction $\xi : X' \to Y$ is a function. We call $\mathsf{dom}\,\xi := X'$ the *domain* and $\mathsf{ker}\,\xi := X - X'$ the *kernel* of ξ. For each $x \in X'$, we write $\xi(x)$ for the unique element $y \in Y$ paired with x, as usual. Similarly, we write $\xi(A)$ for the set of elements $\xi(x)$ with $x \in A \cap X'$. The *image* of ξ is of course the entire reachable set, $\mathsf{im}\,\xi := \xi(X)$. If $\xi : X \rightarrowtail Y$ and $\eta : Y \rightarrowtail Z$ are partial functions, then their *composition* is the partial function $\eta\xi : X \rightarrowtail Z$ consisting of all pairs $(x, z) \in X \times Z$ for which there exists $y \in Y$ such that $y = \xi(x)$ and $z = \eta(y)$. Thus, we have a category of sets and partial functions, which we denote as **Part**. The zero object in this category is the empty set, which is connected to all other sets by empty partial functions. It will be convenient to limit the objects in this category to finite sets.

2.2 Matchings

We call an injective partial function $\alpha : A \rightarrowtail B$ a *matching*. Its restriction to the domain and the image is a bijection, hence the name. Bijections have inverses and so do matchings, namely $\alpha^{-1} : B \rightarrowtail A$ with $(b, a) \in \alpha^{-1}$ iff $(a, b) \in \alpha$.[2] Clearly, the composition of two matchings is again a matching. We therefore have a category, and we write **Mch** for this subcategory of **Part**: its objects are finite sets and its arrows are matchings. Writing $[k] := \{1, 2, \ldots, k\}$, we may assume that $A = [p]$ and $B = [q]$, in which p and q are the cardinalities of A and B. Representing the matching by its matrix, $M = (M_{ij})$, we thus get

[2] To be more precise, we should call it a *weak inverse*, because $\alpha^{-1}\alpha$ and $\alpha\alpha^{-1}$ are identities on the domain and image of α and not necessarily on A and B. We simplify language by ignoring this subtlety.

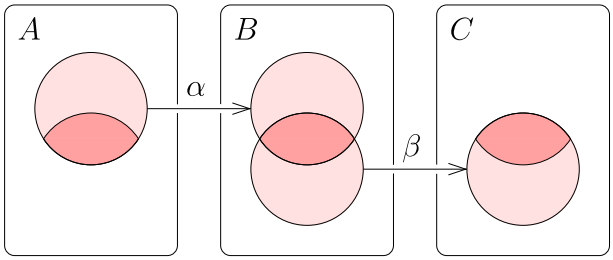

Fig. 2 The composition of two matchings. Its domain and image are the *dark regions* in A and in C

$$M_{ij} = \begin{cases} 1 & \text{if } (j,i) \in \alpha, \\ 0 & \text{otherwise}, \end{cases}$$

for $j \in [p]$ and $i \in [q]$. Matrices of matchings are characterized by having at most one non-zero entry in each row and in each column. It follows that there are equally many non-zero rows (the cardinality of the image) as there are non-zero columns (the cardinality of the domain). The *rank* of the matching is this common cardinality, $\mathsf{rank}\,\alpha := \#\mathsf{dom}\,\alpha = \#\mathsf{im}\,\alpha$. The simple structure of matchings makes it easy to compute the ranks of compositions. Letting $\beta : B \nrightarrow C$ be another matching, the rank of $\beta\alpha : A \nrightarrow C$ is the cardinality of $\mathsf{im}\,\alpha \cap \mathsf{dom}\,\beta$; see Fig. 2.

We can rewrite this as

$$\#(\mathsf{dom}\,\beta - \mathsf{im}\,\alpha) = \mathsf{rank}\,\beta - \mathsf{rank}\,\beta\alpha. \tag{1}$$

It will be useful to extend this relation to the composition of three matchings, adding $\gamma : C \nrightarrow D$ to the two we already have. The number of elements in the domain of β that are neither in the image of α nor map to the domain of γ is

$$\#(\mathsf{dom}\,\beta - \mathsf{im}\,\alpha - \mathsf{dom}\,\gamma\beta) = \mathsf{rank}\,\beta - \mathsf{rank}\,\beta\alpha - \mathsf{rank}\,\gamma\beta + \mathsf{rank}\,\gamma\beta\alpha. \tag{2}$$

To see the second line, we construct a set Ω, first by taking the disjoint union of the sets A, B, C, and D, and second by identifying two elements if they occur in a common pair, which may be in α, β, or γ. After identification, each matching is a subset of Ω, namely $\alpha = A \cap B$, $\beta\alpha = A \cap B \cap C$, etc. Using the identification, the left-hand side of (2) may be rewritten as the cardinality of the set

$$\begin{aligned} &(B \cap C) - (A \cap B) - (B \cap C \cap D) \\ &\quad = (B \cap C) - [(A \cap B \cap C) \cup (B \cap C \cap D)]. \end{aligned} \tag{3}$$

By elementary inclusion–exclusion, its cardinality is the right-hand side of (2).

2.3 Linear Maps

Assuming a fixed field, we now consider the category **Vect**, whose objects are the finite-dimensional vector spaces over this field, and whose arrows are the linear maps

between these vector spaces. The *dimension* of a vector space, U, is of course the cardinality of its basis, which we denote as $\dim U$. Letting $\upsilon : U \to V$ be a linear map, we write $\ker \upsilon := \upsilon^{-1}(0)$ for the *kernel*, $\operatorname{im} \upsilon := \upsilon(U)$ for the *image*, and $\operatorname{rank} \upsilon := \dim \upsilon(U)$ for the *rank* of υ.

Given bases A of U and B of V, we construct the matrix $M = (M_{ij})$ of υ in these bases. In particular, M_{ij} is the coefficient of the ith basis vector of B in the representation of the image of the jth basis vector of A. It is generally not the matrix of a matching because υ does generally not map A to B. However, if M is the matrix of a matching, then the partial function $\alpha : A \nrightarrow B$, consisting of all pairs $(a, b) \in A \times B$ with $\upsilon(a) = b$, is a matching that satisfies $\ker \alpha = \ker \upsilon \cap A$, $\operatorname{im} \alpha = \operatorname{im} \upsilon \cap B$, and, most importantly,

$$\operatorname{rank} \alpha = \operatorname{rank} \upsilon. \tag{4}$$

A matching with this property exists, and we can compute it by reducing M to Smith normal form. However, it is not necessarily unique. On the other hand, any two matchings $\alpha : A \nrightarrow B$ and $\alpha' : A' \nrightarrow B'$ that satisfy (4)—albeit possibly for different bases A, A' of U and B, B' of V— are conjugate in **Mch**. Indeed, we have $\#A = \#A', \#B = \#B'$, and $\operatorname{rank} \alpha = \operatorname{rank} \alpha'$ from (4), which suffices for the existence of bijections that imply the conjugacy of α and α'.

We have a functor $\mathrm{Lin} : \mathbf{Mch} \to \mathbf{Vect}$ which sends a finite set A to the linear space spanned by A and a matching $\alpha : A \to B$ to the linear map defined on a basis vector $a \in A$ as $\alpha(a)$ if a is in the domain of α and as zero otherwise. It follows from the discussion of the preceding paragraph that for every linear map υ in **Vect**, there exists a matching α in **Mch** such that $\upsilon = \mathrm{Lin}(\alpha)$ and any two matchings with this property are conjugate.

2.4 Eigenvalues and Eigenspaces

Still assuming the same field, we consider a vector space, U, and a linear self-map, $\phi : U \to U$. Letting t be an element in the field, we set

$$E_t(\phi) := \{u \in U \mid \phi(u) = tu\}. \tag{5}$$

If $E_t(\phi) \neq 0$, then t is an *eigenvalue* of ϕ, and $E_t(\phi)$ is the corresponding *eigenspace*. As usual, the non-zero elements of $E_t(\phi)$ are referred to as the *eigenvectors* of ϕ and t. It should be clear that $E_t(\phi)$ is a subspace of U and thus a vector space itself. We find it convenient to formalize the transition from the linear self-map to its eigenspaces. To this end, we consider another endomorphism, $\phi' : U' \to U'$, and a linear map $\upsilon : U \to U'$ such that

$$\begin{array}{ccc} U & \xrightarrow{\phi} & U \\ \downarrow{\scriptstyle \upsilon} & & \downarrow{\scriptstyle \upsilon} \\ U' & \xrightarrow{\phi'} & U' \end{array} \tag{6}$$

commutes. We can think of this diagram as an arrow in a new category **Endo**(**Vect**); see e.g. [19]. In particular, the objects in **Endo**(**Vect**) are the endomorphisms in **Vect**, and the arrows are the linear maps that commute with the endomorphisms. Fixing t, we now map ϕ to $E_t(\phi)$ and ϕ' to $E_t(\phi')$, which are objects in **Vect**. Since (6) commutes, the image of an eigenvector in $E_t(\phi)$ belongs to $E_t(\phi')$. This motivates us to define the restriction of υ to $E_t(\phi)$ and $E_t(\phi')$ as the image of υ under E_t, thus completing the definition of E_t as the *eigenspace functor* from **Endo**(**Vect**) to **Vect**.

2.5 Eigenspace Functor for Pairs

The situation in this paper is more general, and we need an extension from endomorphisms to pairs of linear maps. Let $\varphi, \psi : U \to V$ be such a pair, and define $\bar{E}_t(\varphi, \psi) := \{u \in U \mid \varphi(u) = t\psi(u)\}$. This is a subspace of U, but it contains the entire intersection of the two kernels, which we remove by taking the quotient:

$$E_t(\varphi, \psi) := \bar{E}_t(\varphi, \psi) / (\ker \varphi \cap \ker \psi) . \tag{7}$$

Assuming $E_t(\varphi, \psi) \neq 0$, we call t an *eigenvalue* of the pair, and $E_t(\varphi, \psi)$ the corresponding *eigenspace*. The non-zero elements of $E_t(\varphi, \psi)$ are the *eigenvectors* of φ, ψ, and t. Similar to the case of endomorphisms, $E_t(\varphi, \psi)$ is a vector space, although its elements are not the original vectors but equivalence classes of such. To formalize the transition, we consider a second pair $\varphi', \psi' : U' \to V'$ and linear maps υ and ν such that

$$\begin{array}{ccccc} V & \xleftarrow{\varphi} & U & \xrightarrow{\psi} & V \\ \downarrow{\scriptstyle \nu} & & \downarrow{\scriptstyle \upsilon} & & \downarrow{\scriptstyle \nu} \\ V' & \xleftarrow{\varphi'} & U' & \xrightarrow{\psi'} & V' \end{array} \tag{8}$$

commutes. We can think of this diagram as an arrow in a new category as follows. The objects in **Pairs**(**Vect**) are the horizontal pairs of linear maps, and the arrows are the vertical pairs of linear maps that form commutative diagrams, as in (8). Fixing t, we can now map φ, ψ to $E_t(\varphi, \psi)$ and φ', ψ' to $E_t(\varphi', \psi')$, which are objects in **Vect**. Since (8) commutes, the images of the vectors in a class $[u] \in E_t(\varphi, \psi)$ form an equivalence class $[\upsilon(u)] \in E_t(\varphi', \psi')$. We thus define the arrow that maps $[u]$ to $[\upsilon(u)]$ as the image of υ, ν under E_t. This completes the definition of E_t as the *eigenspace functor* from **Pairs**(**Vect**) to **Vect**.

It is easy to see that if the vertical maps in (8) are isomorphisms, then $E_t(\varphi, \psi)$ and $E_t(\varphi', \psi')$ are isomorphic. Starting with (8), suppose now that $V = V' = U'$ and that ν and φ' are identities, and redraw the diagram in triangular form:

$$\begin{array}{ccccc} & & U & & \\ & {\scriptstyle\varphi}\swarrow & & \searrow{\scriptstyle\psi} & \\ V & & \xrightarrow{\psi'} & & V \end{array} \tag{9}$$

If φ is an isomorphism, then so is υ, which implies that $E_t(\varphi, \psi)$ and $E_t(\psi')$ are isomorphic. This little fact will be useful in Sect. 5, when we analyze our algorithm

by comparing eigenspaces obtained from the homology endomorphism induced by a continuous self-map $f : \mathbb{X} \to \mathbb{X}$ and from pairs of linear maps induced in homology by projections from the graph of f (comp. diagram (29)).

3 Towers and Persistence

In this section, we study what can persist along paths in a category, similarly to the very recent, independently obtained results [1]. We show that parallel paths in **Vect** and **Mch** will naturally lead to the concept of persistent homology, now within a more general framework than in the traditional literature. We note that the concept of tower introduced in the sequel corresponds to persistence module in [23]. We prefer to speak about towers to avoid phrases like 'persistent module of persistence modules'.

3.1 Paths and Categories of Paths in a Category

A *tower* is a path with finitely many non-zero objects in a category. More formally, it consists of objects X_i and arrows $\xi_i : X_i \to X_{i+1}$, for all $i \in \mathbb{Z}$, in which all but a finite number of the X_i are the zero object. We denote this tower as $\mathcal{X} = (X_i, \xi_i)$. In later discussions, we will refer to compositions of the arrows, so we write ξ_i^i for the identity arrow of X_i and define

$$\xi_i^j := \xi_{j-1} \dots \xi_{i+1}\xi_i : \ X_i \to X_j, \tag{10}$$

for $i < j$.

Suppose $\mathcal{Y} = (Y_i, \eta_i)$ is a second tower in the same category, and there is a vector of arrows $\varphi = (\varphi_i)$, with $\varphi_i : X_i \to Y_i$, such that $\eta_i\varphi_i = \varphi_{i+1}\xi_i$ for all i. Referring to this vector of arrows as a *morphism*, we denote this by $\varphi : \mathcal{X} \to \mathcal{Y}$. To verify that the towers and morphisms form a new category, we note that the identity morphism is the vector of identity arrows, and that morphisms compose naturally. The zero object is the tower consisting solely of zero objects. Finally, an *isomorphism* is an invertible morphism; it consists of invertible arrows between objects that commute with the arrows of the towers. We remark that arrows and morphisms are alternative terms for the same notion in category theory. We find it convenient to use both so we can emphasize different levels of the construction.

3.2 Persistence in a Tower of Matchings

As a first concrete case, we consider a tower $\mathcal{A} = (A_i, \alpha_i)$ of matchings. Recall that $\mathsf{rank}\,\alpha_i^j$ is the number of pairs in α_i^j. We formalize this notion by introducing the *rank function* $\mathsf{a} : \mathbb{Z} \times \mathbb{N}_0 \to \mathbb{Z}$ defined by $\mathsf{a}_i^j := \mathsf{a}(i, j-i) := \mathsf{rank}\,\alpha_i^j$, where $\mathbb{N}_0$ is the set of non-negative integers. It can alternatively be understood by counting intervals, as we now explain. Letting A be the disjoint union of the A_i, we call a non-empty partial function $a : \mathbb{Z} \nrightarrow A$ an *string* in $\mathcal{A}$ if the domain of a is an interval $[k, \ell]$ in $\mathbb{Z}$, $a_i = a(i)$ belongs to A_i for every $k \le i \le \ell$, and $a_{i+1} = \alpha_i(a_i)$ whenever $k \le i < \ell$; see Fig. 3.

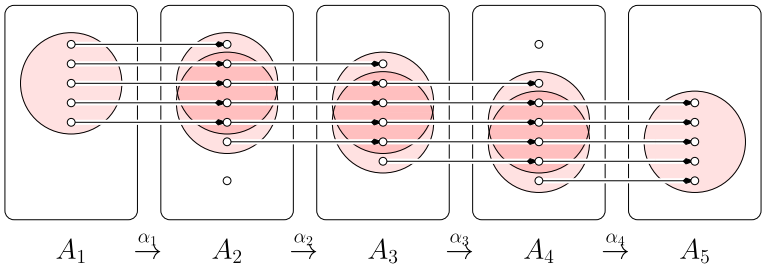

Fig. 3 A tower of matchings with persistence intervals [1, 2], [1, 3], [1, 4], *twice*[1, 5], [2, 2], [2, 5], [3, 5], [4, 4], [4, 5]

The string is *maximal* if it cannot be extended at either end. Specifically, a is maximal iff $a_k \notin \mathsf{im}\,\alpha_{k-1}$ and $a_\ell \in \mathsf{ker}\,\alpha_\ell$. Finally, by a *persistence interval*[3], we mean the domain of a maximal string in $\mathcal{A}$.

It should be clear that a_k^ℓ is the number of strings in $\mathcal{A}$ with domain $[k, \ell]$. Equivalently, it is the number of maximal strings with domains that contain $[k, \ell]$. We can also reverse the relationship and compute the number of persistence intervals from the rank function. Letting $\#_{[i,j]} = \#_{[i,j]}(\mathcal{A})$ denote the number of maximal strings with domain $[i, j]$, we have

$$\#_{[i,j]} = \mathsf{a}_i^j - \mathsf{a}_{i-1}^j - \mathsf{a}_i^{j+1} + \mathsf{a}_{i-1}^{j+1}. \tag{11}$$

To see this, we consider again A and identify elements if they belong to a common pair in any of the α_i. After identification, every point a in A corresponds to a maximal string. The domain of this maximal string is $[i, j]$ iff a belongs to the domain of α_i^j but not to the image of α_{i-1} and not to the domain of α_i^{j+1}. The relation (11) now follows from (2) is applied to $\alpha = \alpha_{i-1}$, $\beta = \alpha_i^j$, $\gamma = \alpha_j$.

This relationship motivates us to introduce the *persistence diagram* of $\mathcal{A}$ as the multiset of persistence intervals, which we denote as $\mathrm{Dgm}(\mathcal{A})$. Note that $\#_{[i,j]}$ is the multiplicity of $[i, j]$. The number of intervals in the persistence diagram, counted with multiplicities, is therefore $\#\mathrm{Dgm}(\mathcal{A}) = \sum_{i \le j} \#_{[i,j]}$. It is important to observe that the persistence diagram characterizes the tower up to isomorphisms.

Equivalence Theorem A *Letting $\mathcal{A}$ and $\mathcal{B}$ be towers in* **Mch**, *the following conditions are equivalent:*

(i) *$\mathcal{A}$ and $\mathcal{B}$ are isomorphic;*
(ii) *the rank functions of $\mathcal{A}$ and $\mathcal{B}$ coincide;*
(iii) *the persistence diagrams of $\mathcal{A}$ and $\mathcal{B}$ are the same.*

Proof (i) $\Rightarrow$ (ii). Since $\mathcal{A}$ and $\mathcal{B}$ are isomorphic, we have invertible arrows $\theta_i : A_i \to B_i$ that commute with the arrows in $\mathcal{A}$ and $\mathcal{B}$. It follows that $\mathsf{rank}\,\alpha_i^j = \mathsf{rank}\,\beta_i^j$, for all $i \le j$.

[3] We note a difference in convention to most of the related literature, in which persistence intervals are defined half-open. In particular, $[k, \ell]$ in our notation corresponds to $[k, \ell + 1)$ in [10]. Reading them as intervals in $\mathbb{Z}$, they are the same.

(ii) $\Rightarrow$ (iii). The rank function determines the multiplicities of the intervals in the persistence diagram by (11).

(iii) $\Rightarrow$ (i). To construct the required isomorphism, we first match up the intervals, and using the matching, we match up the underlying points. □

3.3 Persistence in a Tower of Linear Maps

We return to assuming a fixed field, and consider a tower $\mathcal{U} = (U_i, \upsilon_i)$ in the category of vector spaces over this field.[4] For each i, let A_i be a basis of U_i. Restricting υ_i to A_i and A_{i+1}, we get a partial function $\alpha_i : A_i \nrightarrow A_{i+1}$, again for every i.

Definition *We call the tower of partial functions $\mathcal{A} = (A_i, \alpha_i)$ a* basis *of the tower $\mathcal{U}$ if α_i is a matching and* $\mathsf{rank}\,\alpha_i = \mathsf{rank}\,\upsilon_i$, *for every i.*

We have seen in Sect. 2 that for each $\upsilon_i : U_i \to U_{i+1}$, there are bases A_i and A_{i+1} of the two vector spaces such that the implied partial function $\alpha_i : A_i \nrightarrow A_{i+1}$ is a matching that satisfies $\mathsf{rank}\,\alpha_i = \mathsf{rank}\,\upsilon_i$; see (4). We will show shortly that such bases exist for all vector spaces in the tower simultaneously; see the Basis Lemma below. For now, we assume that $\mathcal{A} = (A_i, \alpha_i)$ is such a basis, deferring the proof to later. Considering compositions α_i^j, we note that $\mathsf{rank}\,\alpha_i^j = \mathsf{rank}\,\upsilon_i^j$. Consequently, the rank functions of $\mathcal{A}$ and $\mathcal{U}$ are the same. The basis of $\mathcal{U}$ is not necessarily unique, but the rank function does not depend on the choice. Thus, we can define the *persistence diagram* of $\mathcal{U}$ as the persistence diagram of a basis, $\mathrm{Dgm}(\mathcal{U}) := \mathrm{Dgm}(\mathcal{A})$. We also write $\#_{[i,j]}(\mathcal{U}) := \#_{[i,j]}(\mathcal{A})$ for the multiplicity of the interval $[i, j]$ in the persistence diagram of $\mathcal{U}$. Writing u_i^j for the rank of υ_i^j, we thus get

$$\#_{[i,j]}(\mathcal{U}) = \mathsf{u}_i^j - \mathsf{u}_{i-1}^j - \mathsf{u}_i^{j+1} + \mathsf{u}_{i-1}^{j+1} \tag{12}$$

from (11). Similarly, we can generalize the Equivalence Theorem A to the case of linear maps between finite-dimensional vector spaces.

Equivalence Theorem B *Letting $\mathcal{U}$ and $\mathcal{V}$ be towers in* **Vect**, *the following conditions are equivalent:*

(i) *$\mathcal{U}$ and $\mathcal{V}$ are isomorphic;*
(ii) *the rank functions of $\mathcal{U}$ and $\mathcal{V}$ coincide;*
(iii) *the persistence diagrams of $\mathcal{U}$ and $\mathcal{V}$ are the same.*

Proof Implications (i) $\Rightarrow$ (ii) and (ii) $\Rightarrow$ (iii) are trivial. To see that (iii) $\Rightarrow$ (i) select bases $\mathcal{A}$ and $\mathcal{B}$, respectively, in $\mathcal{U}$ and $\mathcal{V}$ by the Basis Lemma below. Then $\mathrm{Dgm}(\mathcal{A}) = \mathrm{Dgm}(\mathcal{B})$. Thus, Equivalence Theorem A implies that $\mathcal{A}$ and $\mathcal{B}$ are isomorphic, and the bijections between the bases $\mathcal{A}$ and $\mathcal{B}$ define the requested isomorphisms between $\mathcal{U}$ and $\mathcal{V}$. □

[4] Part of the theory in this section can be developed for the more general case of finitely generated modules over a principal ideal domain. For reasons of simplicity, and because the crucial connection to matchings relies on stronger algebraic properties, we limit this discussion to vector spaces over a field right from the start.

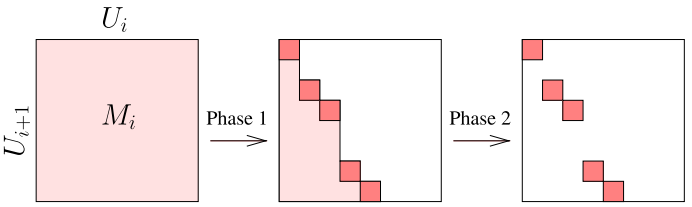

Fig. 4 Two-phase reduction of the matrix. The *shaded areas* contain zeros and non-zeros, the *white areas* contain only zeros, and all *dark squares* are 1

3.4 Tower Bases

We now prove the technical result assumed above to get Equivalence Theorem B. It is not new and can be found in different words and with a less elementary proof in [2,23].

Basis Lemma *Every tower in* **Vect** *has a basis.*

Proof We construct the basis in two phases of an algorithm, as sketched in Fig. 4. Let $\mathcal{U} = (U_i, \upsilon_i)$ be a tower in **Vect**, and for each i, let M_i be the matrix that represents the map υ_i in terms of the given bases of U_i and U_{i+1}.

In Phase 1, we use column operations to turn M_i into column echelon form, as sketched in Fig. 4 in the middle. We get a strictly descending staircase of non-zero entries, with zeros and non-zeros below and zeros above the staircase. Here, we call the collection of topmost non-zero entries in the columns the *staircase*, and we multiply with inverses so that all entries in the staircase are equal to 1. By definition, each column contains at most one element of the staircase, and by construction, each row contains at most one element of the staircase. The reduction to echelon form is done from right to left in the sequence of matrices; that is, in the order of decreasing index i. Indeed, every column operation in M_i changes the basis of U_i, so we need to follow up with the corresponding row operation in M_{i-1}. Since M_{i-1} has not yet been transformed to echelon form, there is nothing else to do.

In Phase 2, we use row operations to turn the column echelon into the normal form, as sketched in Fig. 4 on the right. Equivalently, we preserve the staircase and turn all non-zero entries below it into zeros. To do this for a single column, we add multiples of the row of its staircase element to lower rows. Processing the columns from left to right, this requires no backtracking. The reduction to normal form is done from left to right in the sequence of matrices; that is, in the order of increasing index i. Each row operation in M_i changes the basis of U_{i+1}, so we need to follow up with the corresponding column operation in M_{i+1}. This operation is a right-to-left column addition, which preserves the echelon form. Since M_{i+1} has not yet been transformed to normal form, there is nothing else to do.

In summary, we have an algorithm that turns each matrix M_i into a matrix in which every row and every column contain at most one non-zero element, which is 1. This is the matrix of a matching. Since we use only row and column operations, the ranks of the matrices are the same as at the beginning. Each column operation in M_i has a corresponding operation on the basis of U_i. Similarly, each row operation in M_i has a

corresponding operation on the basis of U_{i+1}. By performing these operations on the bases of the vector spaces, we arrive at a basis of the tower. □

3.5 Persistent Homology and Derivations

Persistent homology as introduced in [11,23] may be viewed as a special case of the persistence of towers of vector spaces. To see this, let $\mathcal{C} = (C_i, \gamma_i)$ be a tower of chain complexes, with γ_i the inclusion of C_i in C_{i+1}, and obtain $\mathcal{H} = (H_i, \eta_i)$ by applying the homology functor. Assuming coefficients in a field, the latter is a tower of vector spaces. The *persistent homology groups* are the images of the η_i^j. A version of persistent homology, recently introduced in [2], studies a sequence of vector spaces U_i and linear maps, some of which go forward, from U_i to U_{i+1}, while others go backward, from U_{i+1} to U_i. This is a *zigzag module* if we have exactly one map between any two contiguous vector spaces. It turns out that the theory of persistence generalizes to this setting. In view of our approach based on matchings, this is not surprising. Indeed, the inverse of a matching is again a matching, so that there is no difference at all in the category of matchings. To achieve the same in the category of vector spaces, we only need to adapt the above algorithm to obtain the zigzag generalization of the Basis Lemma. The adaptation is also straightforward, running the algorithm on a sequence of matrices that are the original matrices for the forward maps and the transposed matrices for the backward maps.

There are several ways one can derive towers from towers, and we discuss some of them. Letting $\mathcal{U} = (U_i, \upsilon_i)$ and $\mathcal{V} = (V_i, \nu_i)$ be towers in **Vect**, we call $\mathcal{V}$ a *subtower* of $\mathcal{U}$ if $V_i \subseteq U_i$ and ν_i is the restriction of υ_i to V_i and V_{i+1}, for each i. Given $\mathcal{U}$ and a subtower $\mathcal{V}$, we can take quotients and define the *quotient tower*, $\mathcal{U}/\mathcal{V} = (U_i/V_i, \varrho_i)$, where ϱ_i is the induced map from U_i/V_i to U_{i+1}/V_{i+1}. Similarly, we can construct towers from a morphism $\varphi : \mathcal{U} \to \mathcal{V}$, where we no longer assume that $\mathcal{V}$ is a subtower of $\mathcal{U}$. Taking kernels and images, we get the *tower of kernels*, which is a subtower of $\mathcal{U}$, and the *tower of images*, which is a subtower of $\mathcal{V}$. Taking the quotients, $U_i/\ker\varphi_i$ and $V_i/\operatorname{im}\varphi_i$, we furthermore define the *towers of coimages* and *of cokernels*. In [7], towers of kernels are used in the analysis of sampled stratified spaces and introduced along with the towers of images and cokernels. The benefit of the general framework presented in this section is that persistence is now defined for all these towers, without the need to prove or define anything else.

Suppose now that $\mathcal{V} = \mathcal{U}$, and let $\phi : \mathcal{U} \to \mathcal{U}$ be an endomorphism. We can iterate ϕ and thus obtain a sequence of endomorphisms. The *generalized kernel* of the component $\phi_i : U_i \to U_i$ is the union of the kernels of its iterated compositions. Similarly, the *generalized image* is the intersection of the images of the iterated compositions:

$$\operatorname{gker}\phi_i := \bigcup_{k=1}^{\infty} \ker \phi_i^{\circ k}, \tag{13}$$

$$\operatorname{gim}\phi_i := \bigcap_{k=1}^{\infty} \operatorname{im}\phi_i^{\circ k}, \tag{14}$$

where $\phi_i^{\circ k}$ is the k-fold composition of ϕ_i with itself. Similar to before, we define two subtowers of $\mathcal{U}$: the *tower of generalized kernels*, denoted as $\mathsf{gker}\,\phi$, and the *tower of generalized images*, denoted as $\mathsf{gim}\,\phi$. By assumption, U_i has finite dimension, which implies that both the generalized kernel and the generalized image are already defined by finite compositions of ϕ_i. Furthermore, the ranks of $\mathsf{gker}\,\phi_i$ and $\mathsf{gim}\,\phi_i$ add up to the rank of U_i. A trivial example of an element in the generalized image is an eigenvector of ϕ_i, but $\mathsf{gim}\,\phi_i$ also contains the eigenvectors of the iterated compositions of ϕ_i and the spaces they span.

Of particular interest are the quotient towers, $\mathcal{U}/\mathsf{gker}\,\phi$ and $\mathcal{U}/\mathsf{gim}\,\phi$, because of their relation to the Leray functor [18] and Conley index theory [8,17]. They may be of interest in the future study of the persistence of the Conley index applied to sampled dynamical systems.

3.6 Tower of Eigenspaces

Of particular interest to this paper is the tower of eigenspaces. When studying the eigenvectors of the endomorphisms, we do this for each eigenvalue in turn. To begin, we note that $\phi : \mathcal{U} \to \mathcal{U}$ is a tower in the category **Endo**(**Vect**). Indeed, each $\phi_i : U_i \to U_i$ is an object, and $\upsilon_i : U_i \to U_{i+1}$ commutes with ϕ_i and ϕ_{i+1}. Applying the eigenspace functor, E_t, we get the tower $\mathcal{E}_t(\phi) = (E_t(\phi_i), \delta_{t,i})$ in **Vect**. Its objects are the eigenspaces, $E_t(\phi_i)$, and its arrows are the restrictions, $\delta_{t,i}$, of the υ_i to $E_t(\phi_i)$ and $E_t(\phi_{i+1})$. We refer to it as the *eigenspace tower* of ϕ for eigenvalue t.

Much of the technical challenge we face in this paper derives from the difficulty in constructing linear self-maps from sampled self-maps. This motivates us to extend the above construction to a pair of morphisms. Let $\mathcal{V} = (V_i, \nu_i)$ be a second tower in **Vect**, let $\varphi, \psi : \mathcal{U} \to \mathcal{V}$ be morphisms between the two towers, and recall that this gives a tower in **Pairs**(**Vect**). Its objects are the pairs $\varphi_i, \psi_i : U_i \to V_i$, and its arrows are the commutative diagrams with vertical maps υ_i and ν_i, as in (8). Similar to the single-map case, we apply the eigenspace functor, E_t, now from **Pairs**(**Vect**) to **Vect**. This gives the tower $\mathcal{E}_t(\varphi, \psi) = (E_t(\varphi_i, \psi_i), \epsilon_{t,i})$ in **Vect**. Its objects are the eigenspaces, and its arrows are the linear maps that map $[u] \in E_t(\varphi_i, \psi_i)$ to $[\upsilon_i(u)] \in E_t(\varphi_{i+1}, \psi_{i+1})$. We refer to it as the *eigenspace tower of the pair* (φ, ψ) for the eigenvalue t. This is the main new tool in our study of self-maps. Of particular interest will be the persistence module of this tower.

4 Algorithm

Assuming a hierarchical representation of an endomorphism, we explain how to compute the persistent homology of its eigenspaces in three steps. The general setting consists of a filtration and an increasing sequence of self-maps. In Step 1, we compute the bases of the two towers obtained by applying the homology functor to the filtrations of spaces and domains. In Step 2, we construct matrix representations of the linear maps in the morphism between the two towers. In Step 3, we compute the eigenvalues and the corresponding eigenspaces as well as their persistence.

4.1 Hierarchical Representation

The algorithm does its computations on a simplicial complex, K, and a partial simplicial map, $\kappa : K \rightarrow\!\!\!\!\!\rightarrow K$. More precisely, κ is a partial map on the underlying space of K, but we will ignore this difference. In addition, we assume a filtration of the complex, $\emptyset = K_0 \subseteq K_1 \subseteq \ldots \subseteq K_n = K$, and of the domain,

$$\mathsf{dom}\,\kappa_0 \subseteq \mathsf{dom}\,\kappa_1 \subseteq \cdots \subseteq \mathsf{dom}\,\kappa_n, \tag{15}$$

with $\kappa_i : K_i \rightarrow\!\!\!\!\!\rightarrow K_i$ being the two-sided restriction of κ to K_i. Writing **Simp** for the category of simplicial complexes and simplicial maps, we use the two filtrations to form a tower in **Pairs**(**Simp**). Its objects are the pairs $\iota_i, \kappa_i' : \mathsf{dom}\,\kappa_i \rightarrow K_i$, in which ι_i is the inclusion and κ_i' is the further restriction of κ_i to the domain. Its arrows are the commuting diagrams connecting contiguous objects by inclusions. Applying the homology functor, we get a tower in **Pairs**(**Vect**), and applying the eigenspace functor, we get a tower in **Vect**. The algorithm in this section computes the persistence diagram of the latter tower.

In principle, it is irrelevant how K and κ are obtained. In the context of sampling an unknown map, we may construct both from a finite sample of that map. We explain this in detail. Write Vert K for the vertex set of K, and let $g : \mathrm{Vert}\,K \rightarrow\!\!\!\!\!\rightarrow \mathrm{Vert}\,K$ be a partial function from the vertex set to itself. If the vertices of a simplex $\sigma \in K$ map to the vertices of a simplex $\tau \in K$, then we extend the vertex map linearly to σ as follows. Letting $x_0, x_1, \ldots, x_p$ be the vertices, and $x = \sum_{i=0}^{p} \lambda_i x_i$ a point of σ, where $\sum_{i=0}^{p} \lambda_i = 1$ and $\lambda_i \geq 0$ for all i, we define

$$\kappa(x) := \sum_{i=0}^{p} \lambda_i g(x_i), \tag{16}$$

which is a point of τ. Doing this for all such simplices σ, we get a partial simplicial map, $\kappa : K \rightarrow\!\!\!\!\!\rightarrow K$. Its domain consists of all simplices whose vertices map to the vertices of a simplex in K. Having constructed κ, it is now easy to construct the partial simplicial maps on the subcomplexes of K. Letting $g_i : \mathrm{Vert}\,K_i \rightarrow\!\!\!\!\!\rightarrow \mathrm{Vert}\,K_i$ be the restriction of g to the vertex set of K_i, we get $\kappa_i : K_i \rightarrow\!\!\!\!\!\rightarrow K_i$ by linear extension as before; it is also the restriction of κ to K_i, as mentioned earlier. We observe that the image is not necessarily the same as the domain. This difference is the reason we construct the tower in **Pairs**(**Simp**) and not in **Endo**(**Simp**), as explained above. Finally, we note that the domains of the κ_i form the filtration (15), as required. Indeed, if the vertices of a simplex σ in K_i map to the vertices of a simplex τ in K_i, then $\sigma \in \mathsf{dom}\,\kappa_j$ for all $i \leq j \leq n$.

Assuming the above hierarchical representation of the sampled map, we explain now how to compute the persistent homology of its eigenspaces in three steps. In Step 1, we compute the bases of the two towers obtained by applying the homology functor to the filtrations of spaces and domains. In Step 2, we construct matrix representations of the linear maps in the morphism between the two towers. In Step 3, we compute the eigenvalues and the corresponding eigenspaces as well as their persistence.

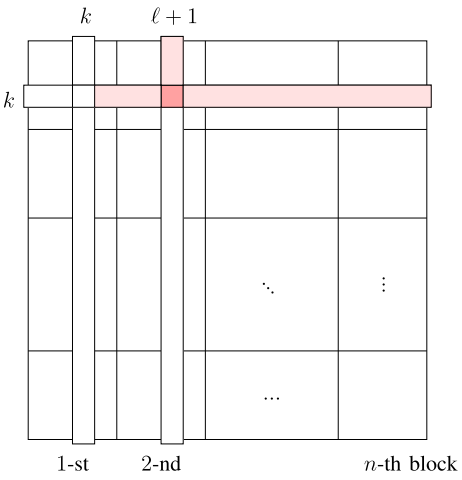

Fig. 5 The *rows* and *columns* of the reduced matrix are decomposed into n blocks each. The kth column is zero, and the lowest non-zero entry in the $(\ell+1)$st column belongs to the kth row. Since the kth and $(\ell+1)$st columns belong to the first and the second blocks, the corresponding persistence interval is $[1, 1]$

Step 1: Spaces Applying the homology functor, we get the tower $\mathcal{X} = (X_i, \xi_i)$ from the filtration of domains $\mathsf{dom}\,\kappa_i$, and the tower $\mathcal{Y} = (Y_i, \eta_i)$ from the filtration of complexes K_i. In this step, we compute the bases of these towers, which we explain for $\mathcal{X}$. Importantly, we represent all domains and maps in a single data structure, and we compute the basis in a single step that considers all maps at once.

Call $\mathsf{dom}\,\kappa_i - \mathsf{dom}\,\kappa_{i-1}$ the *i-th block* of simplices, and sort $\mathsf{dom}\,\kappa$ as $\sigma_1, \sigma_2, \ldots, \sigma_m$ such that each simplex succeeds its faces, and the ith block succeeds the $(i-1)$-st block, for every $1 \le i \le n$. Let D be the ordered boundary matrix of $\mathsf{dom}\,\kappa$; that is, $D[k, \ell]$ is non-zero if σ_k is a codimension-1 face of σ_ℓ, and $D[k, \ell] = 0$, otherwise. The ordering implies that the submatrix consisting of the first i blocks of rows and the first i blocks of columns is the boundary matrix of $\mathsf{dom}\,\kappa_i$, for each i. We use the original persistence algorithm [10, Chap. VII.1] to construct the basis. Similar to the echelon form, it creates a collection of distinguished non-zero entries, at most one per column and row, but to preserve the order, it does not arrange them in a staircase. Specifically, the algorithm uses left-to-right column additions to get D into *reduced form*, which is a matrix R so that the lowest non-zero entries of the columns belong to distinct rows.

Suppose $R[k, \ell+1]$ is the lowest non-zero entry in column $\ell+1$, as in Fig. 5. Then $R[., k] = 0$ and $b_{\ell+1} := R[., \ell+1]$ is the boundary of a chain that contains $\sigma_{\ell+1}$. We note that $b_{\ell+1}$ existed as early as X_k but not earlier, and that it changed to a boundary in $X_{\ell+1}$ but not earlier. In other words, $b_{\ell+1} \notin \mathsf{im}\,\xi_{k-1}$ and $b_{\ell+1} \in \mathsf{ker}\,\xi_\ell$, as required for a maximal string. Assuming σ_k belongs to the ith block of simplices, and $\sigma_{\ell+1}$ belongs to the $(j+1)$-st block, the corresponding persistence interval is $[i, j]$. It is empty if $i = j+1$. In the persistence literature, the above situation is expressed by saying that b_ℓ is *born* at X_i and *dies entering* X_{j+1}. It is also possible that a cycle is born but never dies, in which case we do not have a corresponding lowest non-zero entry in the matrix. But this case can easily be avoided, for example by adding the cone over the entire complex as a last block of simplices to the filtration.

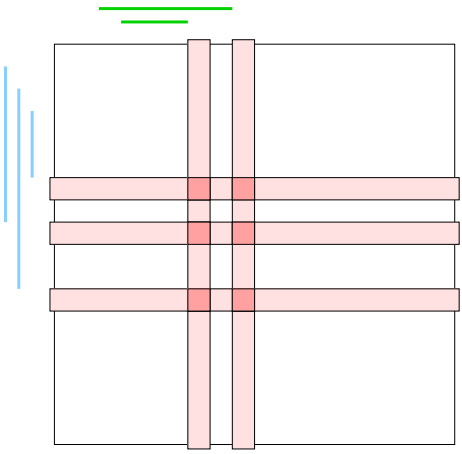

Fig. 6 Extracting the matrix representation of ψ_i from Ψ. For each persistence interval that contains i, we show the *row* or *column* that stores the corresponding cycle

Given an index $1 \leq i \leq n$, we identify the persistence intervals that contain i and get a basis of X_i by gathering the vectors in the corresponding columns of R. The collection of these bases forms a basis of $\mathcal{X}$; see Sect. 3. Running the same algorithm on the filtration of K, we get a basis of $\mathcal{Y}$.

Step 2: Maps Let $\varphi_i, \psi_i : \mathcal{X} \to \mathcal{Y}$ be the morphisms such that φ_i is induced by $\iota_i : \mathsf{dom}\, \kappa_i \to K_i$ and ψ_i is induced by $\kappa_i' : \mathsf{dom}\, \kappa_i \to K_i$. In the second step of the algorithm, we construct matrix representations of the two morphisms. Both matrices, Φ for φ and Ψ for ψ, have their columns indexed by the non-zero columns of the reduced matrix of $\mathcal{X}$ and their rows indexed by the non-zero columns of the reduced matrix of $\mathcal{Y}$. We explain the computations for Ψ.

Letting $R[., \ell+1]$ be a non-zero column in the reduced matrix of $\mathcal{X}$, we recall that $b_{\ell+1} = R[., \ell+1]$ is a cycle in $\mathsf{dom}\, \kappa$. First, we compute its image, $c_{\ell+1} := \kappa(b_{\ell+1})$, which either collapses to zero or is a cycle of the same dimension in $\mathsf{im}\, \kappa$. Second, we write the homology class of $c_{\ell+1}$ as a linear combination of the basis vectors of Y_j, assuming σ_ℓ belongs to the jth block of simplices, as before. Most effectively, this is done as part of the reduction of the boundary matrix of K. Indeed, we can insert the images of the columns into the boundary matrix so that their representation as linear combinations of basis vectors of $\mathcal{Y}$ falls out as a by-product of the reduction. Running the same algorithm for φ, we get the matrix Φ.

The two computed matrices represent the morphisms, φ and ψ, from which the matrices Φ_i and Ψ_i representing the arrows, φ_i and ψ_i, can be extracted. To do so, we first find all persistence intervals that contain i, as before. Second, we collect the intersections of all corresponding rows and columns, as illustrated in Fig. 6.

Step 3: Eigenspaces We recall that Step 2 provides presentations of φ and ψ in terms of the same bases, namely those of $\mathcal{X}$ and of $\mathcal{Y}$ as computed in Step 1. In this step, we compute the filtration of eigenspaces and their persistence, separately for each eigenvalue $t \neq 0$. Fixing the eigenvalue, we compute the eigenspaces of φ_i and ψ_i and we take the quotient relative to the intersection of kernels:

$$\bar{E}_t(\varphi_i, \psi_i) = \ker(\varphi_i - t\psi_i), \tag{17}$$
$$E_t(\varphi_i, \psi_i) = \bar{E}_t(\varphi_i, \psi_i) / (\ker \varphi_i \cap \ker \psi_i). \tag{18}$$

It might be interesting to do the computations incrementally, as in Steps 1 and 2, but here we have to add as well as remove rows and columns, which makes the update operation complicated. Besides, the matrices at this stage tend to be small (see Table 1), so we decide to do the computations for each index i from scratch. At the same time, we extract the kernels of φ_i and ψ_i from Φ_i and Ψ_i, and we use standard methods from linear algebra to compute the quotient. Next, we compute the maps $\xi_i : \mathrm{dom}\,\kappa_i \to \mathrm{dom}\,\kappa_{i+1}$ and their restrictions $\epsilon_{t,i} : E_t(\varphi_i, \psi_i) \to E_t(\varphi_{i+1}, \psi_{i+1})$, thus completing the construction of the eigenspace tower defined by φ and ψ. Finally, we compute the persistence of this tower as explained in Sect. 3.

5 Analysis

Given a finite set of sample points and their images, we apply the algorithm of Sect. 4 to compute information about the otherwise unknown map acting on an unknown space. In this section, we prove that under mild assumptions—about the space, the map, and the sample—this information includes the correct dimension of the eigenspaces. We also show that the persistence diagrams of the eigenspace towers are stable under perturbations of the input.

5.1 Graphs and Distances

Let $f : \mathbb{X} \to \mathbb{X}$ be a continuous map acting on a topological space. For convenience, we assume that $\mathbb{X}$ is a subset of $\mathbb{R}^\ell$, with topology induced by the Euclidean metric.[5] While we are interested in exploring f, we assume that all we know about it is a finite set, $S \subseteq \mathbb{X}$, and the image, $f(s)$, for every point $s \in S$. Assuming that the image of every point is again in S, we write $g : S \to S$ for the restriction of f.[6] The goal is to show that under reasonable conditions, f and g are similar so that we can learn about the former by studying the latter. To achieve this, we need some way to measure distance between two functions whose domains need not be the same. To this end, we consider the graphs of the functions,

$$Gf := \{(x, f(x)) \mid x \in \mathbb{X}\}, \tag{19}$$
$$Gg := \{(s, g(s)) \mid s \in S\}, \tag{20}$$

which are both subsets of $\mathbb{R}^\ell \times \mathbb{R}^\ell$. Using the product metric, the distance between (x, x') and (y, y') in the product space is the larger of the two Euclidean distances,

[5] With occasionally more elaborate formalism, everything we say can be generalized to $\mathbb{X}$ embedded in a general metric space.

[6] In cases in which the image of a point is not in S, we can snap the image to the nearest point in S, which usually implies only a small perturbation of the map. Similarly, we can relax the assumption that g be a restriction of f to allow for errors due to noise, for imprecision of measurement, and for approximations in computation.

$\|x - y\|$ and $\|x' - y'\|$. We compare two maps using the *Hausdorff distance* between their graphs. Recall that the Hausdorff distance between two sets is the infimum radius, r, such that every point of either set has a point at distance at most r in the other set. We note that the Hausdorff distance between the domains of the two functions is bounded from above by the Hausdorff distance between their graphs:

$$\mathrm{Hdf}(\mathbb{X}, S) \leq \mathrm{Hdf}(Gf, Gg), \tag{21}$$

simply because the distance between two points in $\mathbb{R}^\ell \times \mathbb{R}^\ell$ is at least the distance between their projections to the first factor. The Hausdorff distance between two sets is related to the difference between the distance functions they define. To explain this, let $d_{\mathbb{X}}, d_S : \mathbb{R}^\ell \to \mathbb{R}$ be the functions that map each point y to the infimum distance to a point in $\mathbb{X}$ and S, respectively. Similarly, let $d_{Gf}, d_{Gg} : \mathbb{R}^\ell \times \mathbb{R}^\ell \to \mathbb{R}$ be the corresponding distance functions in the product space. Then we have

$$\|d_{\mathbb{X}} - d_S\|_\infty = \mathrm{Hdf}(\mathbb{X}, S), \tag{22}$$

$$\|d_{Gf} - d_{Gg}\|_\infty = \mathrm{Hdf}(Gf, Gg). \tag{23}$$

The conditions under which we can infer properties of f from g include that for small distance thresholds, the sublevel sets of $d_{\mathbb{X}}$ have the same homology as $\mathbb{X}$. To quantify this notion, we assume that $d_{\mathbb{X}}$ is *tame*, by which we mean that every sublevel set has finite-dimensional homology groups, and there are only finitely many *homological critical values* that are the values at which the homology of the sublevel set changes non-isomorphically. Following [6], we define the *homological feature size* of $\mathbb{X}$ as the smallest positive homological critical value of $d_{\mathbb{X}}$, denoting it as $\mathrm{hfs}(\mathbb{X})$. Similarly, we assume that d_{Gf} is tame, and we define $\mathrm{hfs}(Gf)$.

We note that there are functions f for which $\mathrm{hfs}(\mathbb{X}) < \mathrm{hfs}(Gf)$, but there are also functions for which the relation is reversed. For example, the graph of the function that wraps the unit circle in $\mathbb{R}^2$ k times around itself is a curve on a torus in $\mathbb{R}^4$. For large values of k, thickening this curve by a small radius suffices to get the same homotopy type as the torus, while thickening the circle by the same radius does not change its homotopy type.

5.2 Sublevel Sets

If f and g are similar, then the sublevel sets of their distance functions are similar. To make this more precise, we write $A_r := d_A^{-1}[0, r]$, where A may be $\mathbb{X}$, S, Gf, Gg, or some other sets. Then

$$\mathbb{X}_r \subseteq S_{r+\varepsilon} \subseteq \mathbb{X}_{r+2\varepsilon}, \tag{24}$$

$$Gf_r \subseteq Gg_{r+\varepsilon} \subseteq Gf_{r+2\varepsilon}, \tag{25}$$

provided $\varepsilon \geq \mathrm{Hdf}(Gf, Gg)$, which we recall is at least the Hausdorff distance between $\mathbb{X}$ and S. This suggests that we compare f and g based on the sublevel sets of the four distance functions, which is the program we follow.

To apply the algorithms in Sect. 4, we use an indirect approach that encodes the sublevel sets in computationally more amenable simplicial complexes. To explain this connection, we construct a complex by drawing a ball of radius r around each point of S, and let $K_r = K_r(S)$ be the nerve of this collection. It is sometimes referred to as the *Čech complex* of S and r; see [10, Chap. III]. Similarly, we let $L_r = L_r(Gg)$ be the Čech complex of Gg and r.[7] While the complexes are abstract, they are constructed over geometric points, which we use to form maps. Specifically, we write $p_r : L_r \to K_r$ for the simplicial map we get by projecting $\mathbb{R}^\ell \times \mathbb{R}^\ell$ to the first factor, and we write $q_r : L_r \to K_r$ if we project to the second factor. Both are simplicial maps because for every simplex in L_r, its projections to the two factors both belong to K_r. Note that p_r is injective, which implies that its inverse is a partial simplicial map, $p_r^{-1} : K_r \nrightarrow L_r$, and its restriction to the domain is a simplicial isomorphism. Composing it with q_r, we get the partial simplicial map $q_r p_r^{-1} : K_r \nrightarrow K_r$.

We are now ready to relate this construction with the setup we use for our algorithm in Sect. 4. There, we begin with a partial simplicial map, $\kappa : K \nrightarrow K$, and a filtration of K. The filtration is furnished by the sequence of Čech complexes of S, which ends with the complete simplicial complex K over the points in S, and κ is the partial simplicial map defined by $g : S \to S$. In this case, κ happens to be a simplicial map because K is complete. For each radius, r, we have defined $\kappa_r : K_r \nrightarrow K_r$ as the restriction of κ, which is a partial simplicial map. It is not difficult to prove that κ_r is equal to the map we have obtained by composing p_r^{-1} and q_r before. We state this result and its consequence for towers of eigenvalues without proof.

Projection Lemma *Let $\kappa_r : K_r \nrightarrow K_r$ be the partial simplicial map obtained by restricting κ to K_r, and let $p_r, q_r : L_r \to K_r$ be the simplicial maps induced by projecting to the two factors. Then $\kappa_r = q_r p_r^{-1}$, for every $r \geq 0$.*

Recall that κ_r' is the restriction of κ_r to the domain, and $\iota_r : \mathsf{dom}\,\kappa_r \to K_r$ is the inclusion map. In view of the Projection Lemma, we can freely move between the tower of eigenspaces we get for (ι, κ') and (p, q), which we do in the sequel.

5.3 Interleaving

We prepare the main results of this section with a technical lemma about interleaving arrows between eigenspaces. Let $\mathbb{U}, \mathbb{V} \subseteq \mathbb{R}^\ell$, let $h : \mathbb{U} \to \mathbb{U}$ and $k : \mathbb{V} \to \mathbb{V}$ be self-maps, and set $\varepsilon := \mathrm{Hdf}(Gh, Gk)$. Projecting a sublevel set of the distance function of the graph to those of the two factors, we get an object in **Pairs**(**Top**) for h, and another such object for k. Choosing the distance thresholds so they satisfy $r + \varepsilon \leq r'$, we have inclusions and therefore an arrow in **Pairs**(**Top**):

[7] A practically more convenient alternative is the *Vietoris–Rips complex* that consists of all simplices spanned by the data points whose diameters do not exceed $2r$. We will use it for the computations discussed in Sect. 6, but for now we stay with the Čech complex, which has the theoretical advantage that its homotopy type agrees with that of the sublevel set for the same r.

$$\begin{array}{ccccc} \mathbb{U}_r & \xleftarrow{p_r} & Gh_r & \xrightarrow{q_r} & \mathbb{U}_r \\ \downarrow & & \downarrow & & \downarrow \\ \mathbb{V}_{r'} & \xleftarrow{p_{r'}} & Gk_{r'} & \xrightarrow{q_{r'}} & \mathbb{V}_{r'} \end{array} . \tag{26}$$

Applying now the homology functor, we get an arrow in **Pairs**(**Vect**), where we write φ_r, ψ_r and $\upsilon_{r'}, \nu_{r'}$ for the maps induced in homology. Next applying the eigenspace functor, we get the arrow $E_t(\varphi_r, \psi_r) \to E_t(\upsilon_{r'}, \nu_{r'})$ in **Vect**. The technical lemma states two kinds of interleaving patterns, (27) and (28), with the main difference being the reversed direction on the right.

Interleaving Lemma *Let $\mathbb{U}, \mathbb{V} \subseteq \mathbb{R}^\ell$, and $h : \mathbb{U} \to \mathbb{U}, k : \mathbb{V} \to \mathbb{V}$ be such that the associated distance functions are tame. Set $\varepsilon := \mathrm{Hdf}(Gh, Gk)$. If $a + \varepsilon \leq b \leq c \leq d - \varepsilon$, then*

$$\begin{array}{ccc} E_t(\varphi_a, \psi_a) & \to & E_t(\varphi_d, \psi_d) \\ \downarrow & & \uparrow \\ E_t(\upsilon_b, \nu_b) & \to & E_t(\upsilon_c, \nu_c) \end{array} \tag{27}$$

commutes. If $a+\varepsilon \leq b \leq c$ and $a \leq d \leq c-\varepsilon$, then the following diagram commutes:

$$\begin{array}{ccc} E_t(\varphi_a, \psi_a) & \to & E_t(\varphi_d, \psi_d) \\ \downarrow & & \downarrow \\ E_t(\upsilon_b, \nu_b) & \to & E_t(\upsilon_c, \nu_c) \end{array} . \tag{28}$$

Proof We prove (27) using the commutative diagram of inclusions and projections in Fig. 7. Within each disk, we have an object of **Pairs**(**Top**), and for every pair of adjacent disks, we have an arrow of **Pairs**(**Top**).

The vertical arrows are replicas of (26), which are justified by $a + \varepsilon \leq b$ and $c \leq d - \varepsilon$. Each horizontal arrow connects objects in **Pairs**(**Top**) induced by the same self-map, so we only need $a \leq d$ and $b \leq c$, which we get from the assumptions. Applying the homology functor, we get the same diagram, with spaces and maps replaced by the corresponding vector spaces and linear maps. Indeed, the horizontal maps are clear, and the vertical maps are induced by the inclusions that exist because of the assumed relations between a, b, c, d, and ε. Applying now the eigenspace functor, we get the diagram in (27), which is easily seen to commute. The proof for the diagram in (28) is similar and omitted. □

5.4 Small Thickenings

To further prepare our first main result, we recall that the projections from the graph of a continuous map commute with the map itself, a fact best expressed using a commutative diagram:

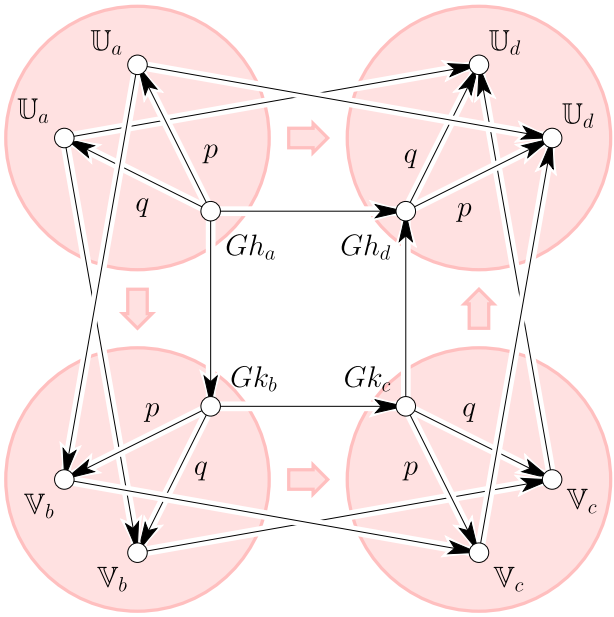

Fig. 7 Sublevel sets of the distance functions and their containment relations. All maps labeled p or q are projections, and all other maps are inclusions

$$\begin{array}{ccc} & Gf, Y & \\ {\scriptstyle p,\mu}\swarrow & & \searrow{\scriptstyle q,\nu} \\ \mathbb{X}, X & \xrightarrow{f,\phi} & \mathbb{X}, X \end{array} \tag{29}$$

Here, we write Y for the homology group of Gf, $\mu : Y \to X$ for the map induced on homology by $p : Gf \to \mathbb{X}$, etc. The restriction of p to Gf and $\mathbb{X}$ is a homeomorphism. We can therefore apply the result stated at the end of Sect. 2, which implies that $E_t(\mu, \nu)$ and $E_t(\phi)$ are isomorphic, for every eigenvalue t. This property persists for small thickenings of $\mathbb{X}$ and Gf assuming the two spaces are compact absolute neighborhood retracts; see [20, p. 290, Thm. 10]. While the name is intimidating, the requirements for a space to be called an absolute neighborhood retract are mild. Since $\mathbb{X}$ and Gf are homeomorphic, the graph is a compact absolute neighborhood retract whenever $\mathbb{X}$ is one. The result about thickening such spaces will be useful in the proof of our first main result, so we state this observation more formally, but without proof.

ANR Lemma *Let $\mathbb{X} \subseteq \mathbb{R}^\ell$ be a compact absolute neighborhood retract, let $f : \mathbb{X} \to \mathbb{X}$ be such that the associated distance functions are tame, and let r be positive but smaller than* $\min\{\mathrm{hfs}(\mathbb{X}), \mathrm{hfs}(Gf)\}$. *Writing μ_r, ν_r for the maps induced in homology by the restrictions of p, q to Gf_r and $\mathbb{X}_r$, the eigenspace $E_t(\mu_r, \nu_r)$ is isomorphic to $E_t(\phi)$, for every eigenvalue t.*

5.5 Convergence

We are now ready to formulate our first main result. As before, we consider a continuous self-map $f : \mathbb{X} \to \mathbb{X}$, and write $\phi : X \to X$ for the endomorphism induced in homology. Let $S \subseteq \mathbb{X}$ be a finite sample of $\mathbb{X}$, and let $g : S \to S$ another map, perhaps the restriction of f to S. As before, we consider the projections from a thickened version of Gg to the two components. Letting φ_r and ψ_r be the corresponding maps induced in homology, we write

$$\epsilon_{t,r}^{r'} : E_t(\varphi_r, \psi_r) \to E_t(\varphi_{r'}, \psi_{r'}) \tag{30}$$

for the map between the eigenspaces. In a nutshell, our result is a relationship between the dimension of $E_t(\phi)$ and the rank of $\epsilon_{t,r}^{r'}$, for special values of r and r'.

Inference Theorem *Let $\mathbb{X} \subseteq \mathbb{R}^\ell$ be a compact absolute neighborhood retract, $S \subseteq \mathbb{X}$, and let $f : \mathbb{X} \to \mathbb{X}$, be a map such that the distance functions for $\mathbb{X}$ and Gf are tame. Then any map $g : S \to S$ satisfies*

$$\dim E_t(\phi) = \operatorname{rank} \epsilon_{t,\varepsilon}^{3\varepsilon}, \tag{31}$$

for all $\operatorname{Hdf}(Gf, Gg) < \varepsilon < \frac{1}{4} \min\{\operatorname{hfs}(\mathbb{X}), \operatorname{hfs}(Gf)\}$.

Proof Recall that $p_r and q_r$ are the projections of Gf_r to the two components, and $\mu_r and \nu_r$ are the maps induced in homology. For $r \leq r'$, we write $\delta_{t,r}^{r'} : E_t(\mu_r, \nu_r) \to E_t(\mu_{r'}, \nu_{r'})$ for the induced map on eigenspaces. In view of the ANR Lemma, it suffices to prove

$$\dim E_t(\mu_r, \nu_r) = \operatorname{rank} \epsilon_{t,\varepsilon}^{3\varepsilon} \tag{32}$$

for $r < \min\{\operatorname{hfs}(\mathbb{X}), \operatorname{hfs}(Gf)$. To prove (32), set $h = f$ and $k = g$ in the diagram in Fig. 7. Applying the eigenspace functor now gives

$$\begin{array}{ccc} E_t(\mu_a, \nu_a) & \xrightarrow{\delta} & E_t(\mu_d, \nu_d) \\ \downarrow & & \uparrow \\ E_t(\varphi_b, \psi_b) & \xrightarrow{\epsilon} & E_t(\varphi_c, \psi_c) \end{array} . \tag{33}$$

The rest of the argument does the accounting for particular choices of the parameters. Let $e > \operatorname{Hdf}(Gf, Gg)$, assume $b \leq c$, and set $a = b - e$ and $d = c + e$. Consider the image of $E_t(\mu_a, \nu_a)$ in $E_t(\mu_d, \nu_d)$. We can either follow the horizontal arrow from left to right, or the down-right-up path, and because of commutativity, we get the same image either way. Dropping the arrows at the end and the beginning of the path does not decrease the dimension of the image. We therefore get inequalities between the dimensions of the images of the horizontal maps, which we now decorate with their parameters:

$$\mathsf{rank}\, \delta_{t,b-e}^{c+e} \leq \mathsf{rank}\, \epsilon_{t,b}^{c}, \tag{34}$$

$$\mathsf{rank}\, \epsilon_{t,b-e}^{c+e} \leq \mathsf{rank}\, \delta_{t,b}^{c}, \tag{35}$$

where we get the second inequality by symmetry, switching the assignments to $h = g$ and $k = f$ in Fig. 7. Next, choose a positive $\eta < \varepsilon$ small enough such that the Hausdorff distance between the graphs of f and g is less than $\varepsilon - \eta$. By assumption, the homological feature sizes of $\mathbb{X}$ and Gf are both larger than 4ε and therefore larger than $4(\varepsilon - \eta)$. Substituting $b = \varepsilon, c = 3\varepsilon, e = \varepsilon - \eta$ in (34), and $b = 2\varepsilon - \eta, c = 2\varepsilon + \eta, e = \varepsilon - \eta$ in (35), we obtain

$$\mathsf{rank}\, \delta_{t,\eta}^{4\varepsilon-\eta} \leq \mathsf{rank}\, \epsilon_{t,\varepsilon}^{3\varepsilon} \leq \mathsf{rank}\, \delta_{t,2\varepsilon-\eta}^{2\varepsilon+\eta}. \tag{36}$$

By definition of ε and η, there are no homological critical values of $d_{\mathbb{X}}$ and d_{Gf} in $[\eta, 4\varepsilon - \eta]$. It follows that the dimensions of the images of $\delta_{t,\eta}^{\eta}$, $\delta_{t,\eta}^{4\varepsilon-\eta}$, and $\delta_{t,2\varepsilon-\eta}^{2\varepsilon+\eta}$ are all the same. Hence,

$$\mathsf{rank}\, \delta_{t,\eta}^{\eta} = \mathsf{rank}\, \epsilon_{t,\varepsilon}^{3\varepsilon}. \tag{37}$$

Since $\mathsf{rank}\, \delta_{t,\eta}^{\eta} = \dim E_t(\mu_\eta, \nu_\eta)$, we see that (32) is satisfied for $r = \eta$. □

The Inference Theorem may be interpreted as a statement of convergence of our algorithm: if the sampling is fine enough, then we are guaranteed to get the dimensions of the eigenspaces as dimensions of persistent homology groups.

5.6 Stability

Next, we strengthen the convergence result and prove the stability of the persistence diagrams of the eigenspace towers under perturbations of the input. This is interesting because we may sample the same self-map twice and wonder what we can say about the relationship between the two results. Most of the work that allows us to give a meaningful answer to this question has already been done. To set the stage, we consider two self-maps, $h : \mathbb{U} \to \mathbb{U}$ and $k : \mathbb{V} \to \mathbb{V}$, in which both $\mathbb{U}$ and $\mathbb{V}$ are embedded in $\mathbb{R}^\ell$. As before, we assume that the distance functions, $d_{\mathbb{U}}, d_{\mathbb{V}}, d_{Gh}, and d_{Gk}$ are tame. We can now form towers in **Pairs**(**Top**) consisting of projections from the sublevel sets of d_{Gh} and d_{Gk} to the sublevel sets of $d_{\mathbb{U}}$ and $d_{\mathbb{V}}$; see Fig. 7. To formalize the result, we define the *bottleneck distance* between two persistence diagrams as the maximum distance between pairs in an optimal bijection:

$$\mathrm{Bot}(\mathsf{E}, \mathsf{F}) = \inf_{\iota:\mathsf{E}\to\mathsf{F}} \max_{P\in\mathsf{E}} \|P - \iota(P)\|_\infty. \tag{38}$$

Here, $P = [a_\mathrm{b}, a_\mathrm{d})$ is a persistence interval in E, now using the original convention in which a_b and a_d are the birth- and death-values. If $Q = [c_\mathrm{b}, c_\mathrm{d})$ is another persistence interval, then we compute $\|P - Q\|_\infty = \max\{|a_\mathrm{b} - c_\mathrm{b}|, |a_\mathrm{d} - c_\mathrm{d}|\}$, as for points in

the plane.[8] Letting $\mathcal{E}_t$ and $\mathcal{F}_t$ be the towers of eigenspaces in **Vect** we get for h and k, we write $\mathrm{Dgm}(\mathcal{E}_t)$ and $\mathrm{Dgm}(\mathcal{F}_t)$ for their persistence diagrams.

Stability Theorem *Let* $\mathbb{U}, \mathbb{V} \subseteq \mathbb{R}^\ell$, *and* $h : \mathbb{U} \to \mathbb{U}$ $k : \mathbb{V} \to \mathbb{V}$ *such that the associated distance functions are tame. Then*

$$\mathrm{Bot}(\mathrm{Dgm}(\mathcal{E}_t), \mathrm{Dgm}(\mathcal{F}_t)) \leq \mathrm{Hdf}(Gh, Gk). \tag{39}$$

Proof According to [4, Thm. 4.9], we only need to verify the ε-strong interleaving of the two towers for ε equal to the Hausdorff distance between Gh and Gk, but this is guaranteed by the Interleaving Lemma. □

Letting h and k be finite samples of $f : \mathbb{X} \to \mathbb{X}$, the Stability Theorem implies that the information they convey about the given function cannot be arbitrarily different. Setting $h = f$ and $k = g$, the theorem quantifies the extent to which the persistence diagram for the sampled points can deviate from that of the original self-map.

6 Experiments

In this section, we present the results of a small number of computational experiments based on the implementation of the algorithm described in Sect. 4. We begin with a brief review of the algorithm and a discussion of the design decisions used in writing the software.

6.1 Implementation

We implement the three steps of the algorithm using the `C++` language and methods from its `STL` library. As mentioned before, we use Vietoris–Rips complexes as the basis of persistence computations, as opposed to the theoretically more satisfying but computationally more expensive Čech complexes. With each simplex, we store the vertices so that their images can be easily computed. Conversely, we use a `map` in `STL` to find the simplex for a given set of vertices. In addition, we compute the kernels and the quotients in Step 3 using methods from the `CAPD` library [24] and the `CAPD::RedHom` library [25].

Step 1 compute the bases of the towers $\mathcal{X} = (X_i, \xi_i)$ and $\mathcal{Y} = (Y_i, \eta_i)$ defined by the filtrations of the domains and complexes. We use the original persistence algorithm with a sparse-matrix representation in which the non-zero elements of each column are stored in a data structure referred to as a `vector` in `C++`.

Step 2 compute the matrix representations of the morphisms $\varphi, \psi : \mathcal{X} \to \mathcal{Y}$. Similar to Step 1, the algorithm works by incremental reduction of two matrices. The result is a compact representation of the maps $\epsilon_{t,i}$ in the eigenspace tower.

[8] Following [6], we assume that each persistence diagram contains copies of all empty intervals—points of the form (a, a)—which are used to complete a bijection or decrease the maximum distance.

Table 1 Time in seconds for constructing the 2-skeleton of the Vietoris–Rips complex and executing the steps of our algorithm for one eigenvalue

No. of points	Skeleton	Step 1	Step 2	Step 3
40	0.14	0.39	0.00	0.04
60	0.67	2.12	0.00	0.15
80	2.21	6.84	0.01	0.39
100	5.18	23.49	0.01	0.86
120	11.13	63.09	0.01	1.50
140	19.53	137.86	0.03	2.18

Step 3 construct the sequence of eigenspaces and compute the corresponding persistence diagram. Since the matrices representing the maps $\epsilon_{t,i}$ tend to be small and dense, we use their full representations and the algorithm in the proof of the Basis Lemma.

All experiments are conducted with an Intel Core2 Quad 2.66 GHz processor with 8 GB RAM, but using only one core. To convey a feeling for the performance of the software, Table 1 states the time needed to process datasets of size between 40 and 140 points, giving complexes between 10,000 and 460,000 simplices.

We mention that the running time can be further improved. In particular, the current implementation is generic, working for any field of coefficients, and the code implementing Step 1 has not yet been optimized. Note the dramatic drop in the running time from Step 1 to Step 2. The reason is the surprisingly small numbers of generators needed in the construction of the matrices Φ and Ψ. In the first set of experiments, we get between 3 and 21 generators for the filtration of $\mathsf{dom}\,\kappa$ and between 5 and 24 generators for the filtration of K. Compare this with the 10,700 to 457,450 simplices in the 2-skeleta of the Vietoris–Rips complex which have to be processed in Step 1. The code for Step 3 takes more time than for Step 2 because it executes computationally demanding procedures in linear algebra.

6.2 Expansion

In our first set of computational experiments, we consider the unit circle in the complex plane, and the function $f : \mathbb{S}^1 \to \mathbb{S}^1$ defined by $f(z) := z^2$. It maps each point on the circle to the point with twice the angle. The 1-dimensional homology of the circle has rank 1, with the circle itself being a generator. Under f, the image of this generator is the circle that wraps around $\mathbb{S}^1$ twice. We see that the map expands the space, doubling the angle between any two points. Our main interest is to see whether the methods of this paper can detect this simple fact.

We chose values for three parameters to generate the datasets on which we run our software: the order of the cyclic field, $\mathbb{Z}_k$ with $k = 1009$, the number of points, $m = 100$, and the width of the Gaussian noise, $\sigma \in [0.00, 0.30]$. The finite field is used because we lack a general algorithm for eigenvalues; instead, we try out all possible values. The sample of the function f is computed by picking points $z_j := \cos(\frac{2j\pi}{m}) + \mathbf{i}\sin(\frac{2j\pi}{m})$, for $0 \leq i < m$, where $\mathbf{i}$ is the imaginary unit. Next, we let

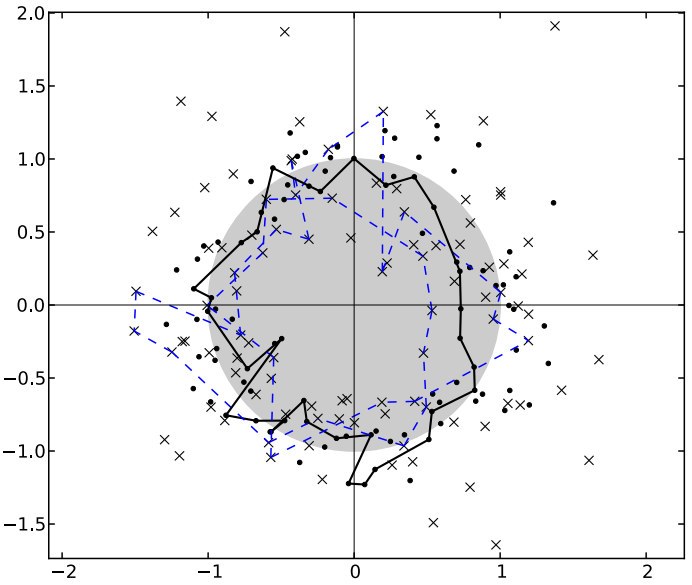

Fig. 8 The representation of the function $f(z) := z^2$ using $m = 100$ points with Gaussian noise $\sigma = 0.18$. The *dots* mark the points in S and the *crosses* mark their squares. The *solid polygon* generates the 1-cycle visible in the persistence diagram of the eigenspace tower for eigenvalue $t = 2$; its image is drawn in *blue dashed lines* and wraps twice around the origin, as expected (Color figure online)

x_j be a point randomly chosen from the isotropic Gaussian kernel with center z_j and width σ. Let S be the set of points x_j. Finally, we set the image of x_j to the point in S that is closest to x_j^2 under the Euclidean metric in the plane. For an example, see Fig. 8. The m points define $n \leq \binom{m}{2}$ different distances and therefore $n + 1$ different Vietoris–Rips complexes. We are only interested in the 1-dimensional homology, so we can limit ourselves to the 2-skeleta of these complexes. To construct them, we use the algorithm in [22] to compute the complete 2-complex over S. Sorting the edges by length, we get the filtration $K_0 \subseteq K_1 \subseteq \ldots \subseteq K_n$. Figure 9 shows the 1-dimensional persistence diagrams thus obtained for four different values of the width σ.

As expected, the persistence of the interval decreases as the noise increases. For $\sigma = 0.30$, we get a low-persistence interval for every value of t. While we do not observe this all the time, this is a generic phenomenon, and we will shed light on it shortly. For now we just mention that the occurrence of every field value as eigenvalue indicates that we do not have sufficient data to see the features of the map.

6.3 Reflection

In our second set of computational experiments, we let $f : \mathbb{S}^1 \to \mathbb{S}^1$ be defined by $f(z) := \bar{z}$, where $\bar{z} = a - \mathbf{i}b$ if $z = a + \mathbf{i}b$. Going around the circle once, in a counterclockwise order, the image under f goes around the circle once in a clockwise order. Again we are interested whether the methods in this paper can detect this fact. The points and their images are chosen in the same way as before, except that the

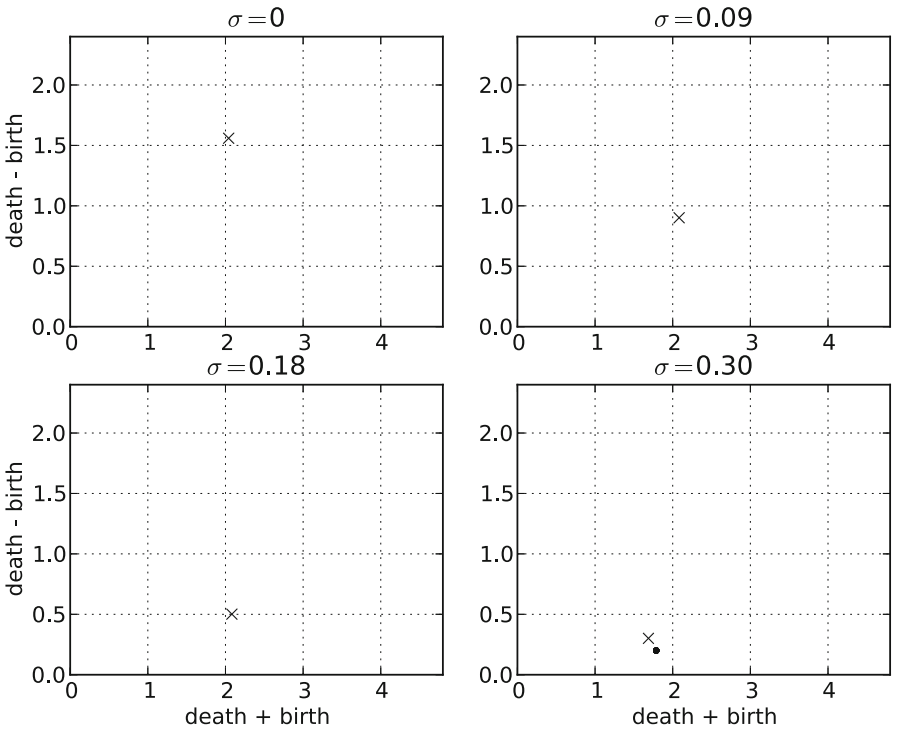

Fig. 9 The 1-dimensional persistence diagrams of $f(z) := z^2$ obtained for data that samples the function with Gaussian noise. *Crosses* mark the persistence intervals for eigenvalue $t = 2$. The *dots* alert us of the fact that we see this interval for all values, not just for $t = 2$

image of x_j is chosen to be the closest point to $\bar{x}_j$. Figure 10 shows the result for $\sigma = 0.27$.

Instead of showing the individual 1-dimensional persistence diagrams, we superimpose them into one diagram. To further facilitate the comparison with the first set of experiments, we draw the superimposed diagrams side by side in Fig. 11. In both cases, the Gaussian noise varies between 0.00 and 0.30. We limit the comparison to the eigenvalues $t = 2$ for the expansion, and $t = -1$ (the inverse of 1 in the used finite field) for the reflection. The diagrams clearly show that the persistence interval shrinks with increasing noise. Indeed, the birth-coordinate grows and the death-coordinate shrinks, so that the sum stays approximately constant, with a faint tendency to shrink. We also see that for the larger noise levels, there are sometimes spurious persistence intervals.

6.4 Abundance of Eigenvalues

We wish to shed light on the phenomenon that for some datasets and some complexes in the filtration, every field value is an eigenvalue of the pair of linear maps. While it might be surprising at first, there is an elementary explanation that has to do with computing the eigenvalues for a pair instead of a single map.

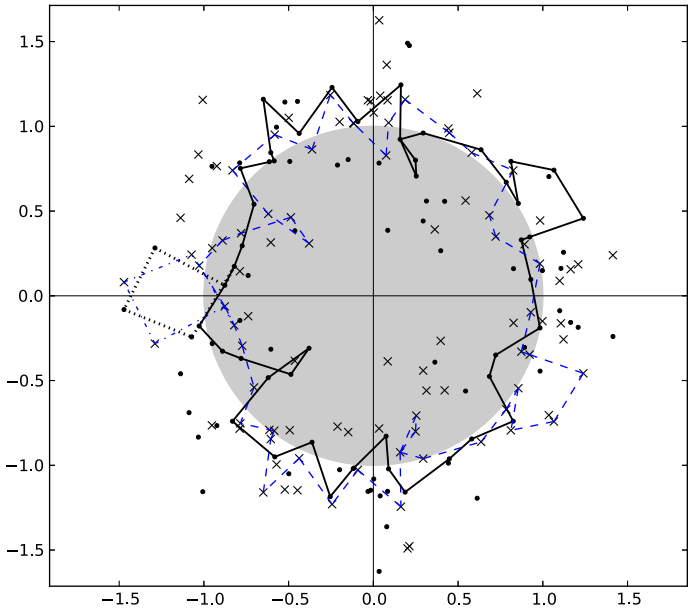

Fig. 10 The representation of the function $f(z) := \bar{z}$ using $m = 100$ points with Gaussian noise $\sigma = 0.27$. The *solid polygon* generates the 1-cycle visible in the persistence diagram of the eigenspace tower for eigenvalue $t = -1$; its image is drawn in *blue dashed lines*. The *dotted polygon* generates a second (spurious) 1-cycle in the same diagram. Correspondingly, there are two persistence intervals, drawn as *crosses* in Fig. 11 (Color figure online)

Here is an illustrative example. Let Y_r be generated by two loops, A and B, and let X_r be generated by a single loop, C. Suppose also that φ_r maps both A and B to C, while ψ_r maps A to C and B to $2C$. Setting $y = iA + jB$, we have $\varphi_r(y) = (i + j)C$ and $\psi_r(y) = (i + 2j)C$. Elementary number theoretic considerations show that for every $t \in \mathbb{Z}_k$, there are i and j such that $t = \frac{i+j}{i+2j}$. In other words, we can find i and j such that $\varphi_r(y)$ is the t-fold multiple of $\psi_r(y)$. Intersecting the two kernels, we get $i + j = i + 2j = 0$ and therefore $i = j = 0$. Hence, taking the quotient has no effect, implying that $E_t(\varphi_r, \psi_r)$ has non-zero rank for every t. Indeed, because the two loops in Y_r map to different multiples of the same loop in X_r, we have enough flexibility to form combinations whose images under the two maps are arbitrary multiples of each other. This can also happen for an endomorphism $\phi_r : Y_r \to Y_r$, for example by setting $C = B$, but in this case we do not have a second map to compare and therefore get $t = 2$ as the only non-zero eigenvalue.

Let us now look at the linear algebra of the situation. In Step 2, we compute matrices Φ_r and Ψ_r representing $\varphi_r, \psi_r : Y_r \to X_r$, and in Step 3, we compute the nullspace of $\Phi_r - t\Psi_r$. The entries of this matrix are degree-1 polynomials in t. Let t_0 be a value at which the matrix reaches its maximum rank, which we denote as k_0. Clearly, $k_0 \leq \min\{\#\text{rows}, \#\text{columns}\}$. Note that $\Phi_r - t_0\Psi_r$ has a full-rank minor of size k_0 times k_0. Let $\Delta(t)$ be the determinant of that minor, but now for arbitrary values t. It is a polynomial of degree k_0, and because $\Delta(t_0) \neq 0$, it is not identically zero and therefore has at most k_0 roots. By choice of t_0, this implies that the matrix has maximum rank for

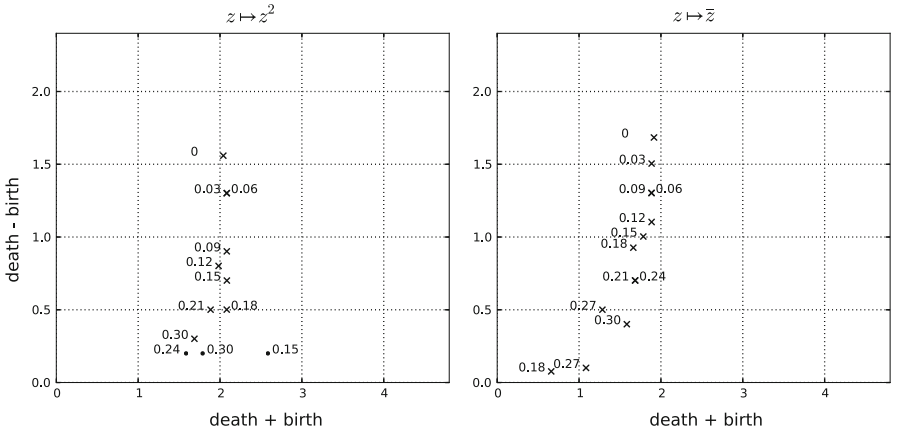

Fig. 11 The superposition of the 1-dimensional persistence diagrams for $f(z) := z^2$ and $t = 2$ on the *left* and for $f(z) := \bar{z}$ and $t = -1$ on the *right*. The *crosses* are labeled by the level of the Gaussian noise used to generate the datasets. The *dots* are labeled similarly, but they alert us of the fact that these persistence intervals occur for all values of t

all but at most k_0 values of t. Correspondingly, the nullspace has minimum dimension, #columns $- k_0$, for all but at most k_0 values of t. This is the dimension of $\bar{E}_t(\varphi_r, \psi_r)$. We still take the quotient by dividing with $\ker \varphi_r \cap \ker \psi_r$, which amounts to reduce the dimension by the dimension of that intersection, which we denote as k_1. The resulting dimension of the nullspace is the same for all but at most k_0 values of t, namely #columns $- k_0 - k_1$. If $k_1 <$ #columns $- k_0$, then $E_t(\varphi_r, \psi_r)$ has positive rank for every value of t. This is what happens for the expanding datasets generated with width $\sigma = 0.15, 0.24, and 0.30$ and for the persistence intervals represented by the dots in the left diagram in Fig. 11. In all other cases, we have $k_1 =$ #columns $- k_0$.

In conclusion, we mention that the extension of the eigenvalue problem to pairs of linear maps for not necessarily square matrices is not well understood. A relevant unpublished manuscript is [5], in which properties of the solution are discussed and a reduction algorithm is given.

6.5 Torus Maps

We test the method to see if it can distinguish between some torus maps. We identify $\mathbb{S}^1$ with the quotient space $\mathbb{R}/\mathbb{Z}$ and consider the torus $\mathbb{T} := \mathbb{S}^1 \times \mathbb{S}^1 \cong \mathbb{R}^2/\mathbb{Z}^2$, equipped with the metric induced by the Euclidean distance in the plane. Each 2-by-2 matrix with integer entries induces a self-map on the torus. Using the cycles $\mathbb{S}^1 \times 0$ and $0 \times \mathbb{S}^1$ as a basis, the same matrix represents the map induced on first homology. We consider the following three matrices,

$$A_1 := \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}, \quad A_2 := \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \quad A_3 := \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}.$$

Note that A_1 is diagonal with non-trivial eigenspace for the eigenvalue 2, which is spanned by two eigenvectors. Matrix A_2 is non-diagonal with diagonal Jordan form.

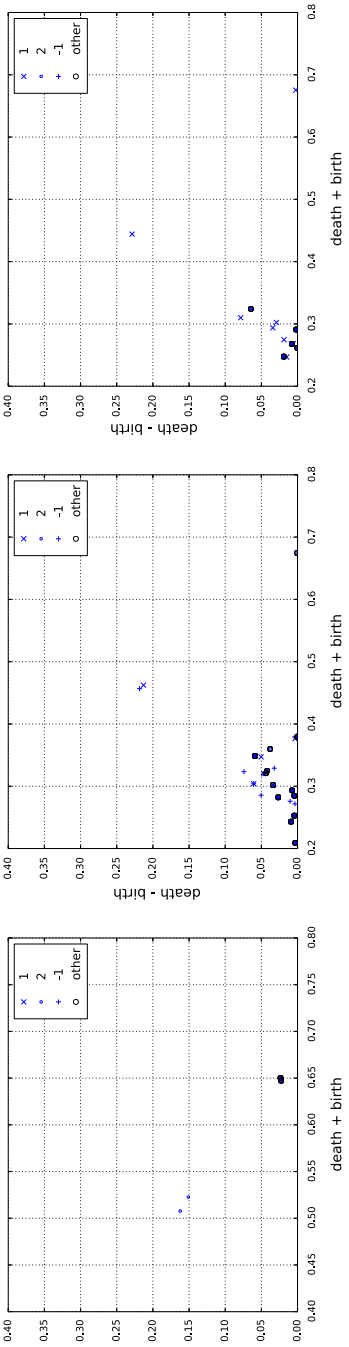

Fig. 12 From *left* to *right*: the 1-dimensional persistence diagrams for f_1, f_2, f_3, each a self-map on the torus

It has one-dimensional eigenspaces for the eigenvalues 1 and -1. Matrix A_3 has non-diagonal Jordan form and has only one non-trivial, one-dimensional eigenspace.

We denote the corresponding self-maps by f_1, f_2, and f_3. Selecting a set of 200 points uniformly at random on the torus, we compute approximations of the f_i. The persistence diagrams of the eigenspaces of the maps induced in first homology are shown in Fig. 12. In each case, the expected eigenvectors clearly separate from the rest of the diagram, facilitating the easy distinction between the three maps.

7 Discussion

The main concept introduced in this paper is the eigenspace tower of a filtered self-map. Together with its persistence diagram, it forms a powerful tool in the study of discretely sampled dynamical systems. Besides the mathematical development, which is based on a category theory approach to persistence, we give an algorithm for computing persistence diagrams of eigenspace towers, and we provide evidence of its efficacy by presenting results obtained with a software implementation. The work reported in this paper raises a number of yet unanswered questions.

- Can the persistence of the eigenspace towers of a pair of morphisms be computed directly, for all eigenvalues simultaneously? To answer this question, we may have to study the persistence of generalized eigenspaces and Jordan form representations of endomorphisms.
- The category approach to persistence opens the door to a number of derived towers, including generalized kernels and generalized images. How can we use their persistence to enhance our understanding of discretely sampled dynamical systems?

We remark that the traditional notion of a discrete dynamical system discretizes time but not space. In contrast, discretizations of space are needed for rigorous numerics of dynamical systems; see e.g. [14, Sect. 10.6]. There, the space is divided into boxes and estimates of the images of the boxes are used as input for the algorithm that computes topological invariants. This is different from the more radical discretization suggested by the work in this paper where we sample the dynamical system only in a finite collection of points. Such an approach may be useful when the dynamical system is available only via experiments. It may also be used to replace rigor in high-dimensional problems where reasonable rigorous estimates are not possible. We believe that it is interesting to continue the program started here and embed persistence more comprehensively in discrete approaches to dynamical systems.

Acknowledgments This research is partially supported by the TOPOSYS project FP7-ICT-318493-STREP, by ESF under the ACAT Research Network Programme, by the Russian Government under mega project 11.G34.31.0053, and by the Polish National Science Center under Grant No. N201 419639.

References

1. P. Bubenik and J.A. Scott. Categorification of persistent homology. Discrete Comput. Geom. 51 (2014), 600–627.
2. G. Carlsson and V. de Silva. Zigzag persistence. Found. Comput. Math. 10 (2010), 367–405.
3. G. Carlsson, T. Ishkanov, V. de Silva and A. Zomorodian. On the local behavior of spaces of natural images. Internat. J. Comput. Vision 76 (2008), 1–12.
4. F. Chazal, D. Cohen-Steiner, M. Glisse, L. J. Guibas and S. Y. Oudot. Proximity of persistence modules and their diagrams. In "Proc. 25th Sympos. Comput. Geom., 2009", 237–246.
5. S. Cohen and C. Tomasi. Systems of bilinear equations. Techn. Rept., Dept. Comput. Sci., Stanford Univ., California, 1997.
6. D. Cohen-Steiner, H. Edelsbrunner and J. Harer. Stability of persistence diagrams. Discrete Comput. Geom. 37 (2007), 103–120.
7. D. Cohen-Steiner, H. Edelsbrunner, J. Harer and D. Morozov. Persistent homology for kernels, images, and cokernels. In "Proc. 20th Ann. ACM-SIAM Sympos. Discrete Alg., 2009", 1011–1020.
8. C. Conley. Isolated Invariant Sets and the Morse Index. Amer. Math. Soc., CBMS Regional Conf. Series Math. 38, 1978.
9. H. Derksen and J. Weyman. Quiver representations. Notices Amer. Math. Soc. 52 (2005), 200–206.
10. H. Edelsbrunner and J. L. Harer. Computational Topology. An Introduction. Amer. Math. Soc., Providence, Rhode Island, 2010.
11. H. Edelsbrunner, D. Letscher and A. Zomorodian. Topological persistence and simplification. Discrete Comput. Geom. 28 (2002), 511–533.
12. J. Gamble and G. Heo. Exploring uses of persistent homology for statistical analysis of landmark-based shape data. J. Multivar. Analysis 101 (2010), 2184–2199.
13. A. Gyulassi, V. Natarajan, V. Pascucci, P.-T. Bremer and B. Hamann. A topological approach to simplification of three-dimensional scalar fields. IEEE Trans. Visual. Comput. Graphics (2006), 474–484.
14. T. Kaczynski, K. Mischaikow and M. Mrozek. Computational Homology. Springer-Verlag, New York, New York, 2004.
15. S. Kleene. Introduction to Meta-Mathematics. North Holland, Amsterdam, the Netherlands, 1952.
16. S. MacLane. Categories for the Working Mathematician. Springer-Verlag, New York, New York, 1971.
17. K. Mischaikow, M. Mrozek (2002) The Conley index theory. In: B. Fiedler, G. Iooss, N. Kopell (eds) Handbook of Dynamical Systems III: Towards Applications. Elsevier, Singapore 393–460
18. M. Mrozek. Leray functor and the cohomological Conley index for discrete time dynamical systems. Trans. AMS 318 (1990), 149–178.
19. M. Mrozek. Normal functors and retractors in categories of endomorphisms. Univ. Jagiell. Acta. Math. 29 (1992), 181–198.
20. E. Spanier. Algebraic Topology. McGraw-Hill or Springer-Verlag, New York, 1966.
21. S. Weinberger. What is ... persistent homology? Notices Amer. Math. Soc. 58 (2011), 36–39.
22. A. Zomorodian. Fast construction of the Vietoris-Rips complex. Computer and Graphics 34 (2010), 263–271.
23. A. Zomorodian and G. Carlsson. Computing persistent homology. Discrete Comput. Geom. 33 (2005), 249–274.
24. CAPD: computer assisted proofs in dynamics. http://www.capd.ii.uj.edu.pl.
25. CAPD::RedHom: reduction algorithms for homology computation. http://www.redhom.ii.uj.edu.pl.